Session 508

Java Graphics

John Burkey
Java Classes Team Technical Lead

What We Will Cover

- Mac OS X Java (Features Overview)
- Tips and Techniques
- Update: Mac OS X Java DP1
  - Promises kept
  - Architecture
- Q&A

Mac OS X

Finally We Have Java2!

Overview of Java Graphics

- PDF rendering
- Object API on top of Quartz
- Same API for rendering on
  - Screen, Images, Printing

High-level Features

- GeneralPath/Shapes
- Custom Fills (textures, gradients)
- Custom Strokes (dashes, widths)
- Custom Compositing
  - (You define the blit)
- Completely portable code

Demo

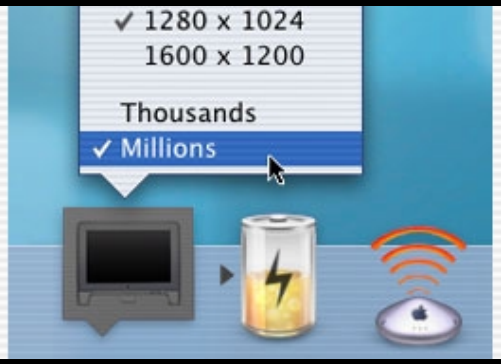
Java2D

Gerard Ziemski
Java Classes Team

Mac OS X Bonuses

- Quartz Rendering
  - The native graphics system
  - Much easier to use than via Carbon, Cocoa
    - No raw Quartz function calling
    - All via easy garbage collected objects
    - We also get plug in Compositing, Strokes…
- Print to PDF
- More in a minute

Performance



Performance: Things You Can Do

- Image creation
- Rendering hints
- Double buffering
- Live window resizing

Image Creation

- Use built in Factory methods
  - GraphicsEnvironment.createCompatibleImage()
- Avoid using non standard image types
- TYPE_CUSTOM, TYPE_INT_BGR ,TYPE_3BYTE_BGR ,TYPE_4BYTE_ABGR ,TYPE_4BYTE_ABGR_PRE ,TYPE_USHORT_565_RGB ,TYPE_BYTE_GRAY ,TYPE_USHORT_GRAY ,TYPE_BYTE_BINARY ,TYPE_BYTE_INDEXED

Code

Image Creation

John Burkey

Rendering Hints (sun.awt.SunHints)

- RENDERING
- ANTIALIASING
- TEXT_ANTIALIASING
- FRACTIONALMETRICS
- DITHERING
- INTERPOLATION
- ALPHA_INTERPOLATION
- COLOR_RENDERING

Code

Rendering Hints

John Burkey

Double Buffering

- Don’t double buffer
- Mac OS X double buffers all Carbon Windows
- We take care of Swing

Mac OS X Live Window Resizing

- Make Component.paint() fast
- If you need to:
  - Do a multi pass render
    - Quick first pass (low detail)
    - Use a render thread for more detail

A Few More

- All graphics objects are created equal
- The more complicated the clip, the slower the graphics speed
- Watch out for too many repaints()
- Profile, then
  - Cache expensive to create objects
    - FontMetrics, possibly some images

Java Update: Promises Kept

Cutting edge technology

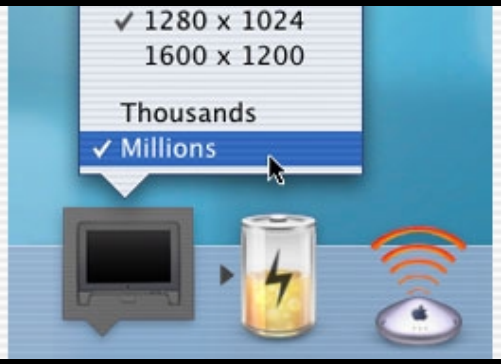
Correctness

Performance

Java Update: Promises Kept

- Much better Quality
  - Heavyweight AWT
  - Swing
  - Printing
  - All image formats
  - Many bugs died this Spring

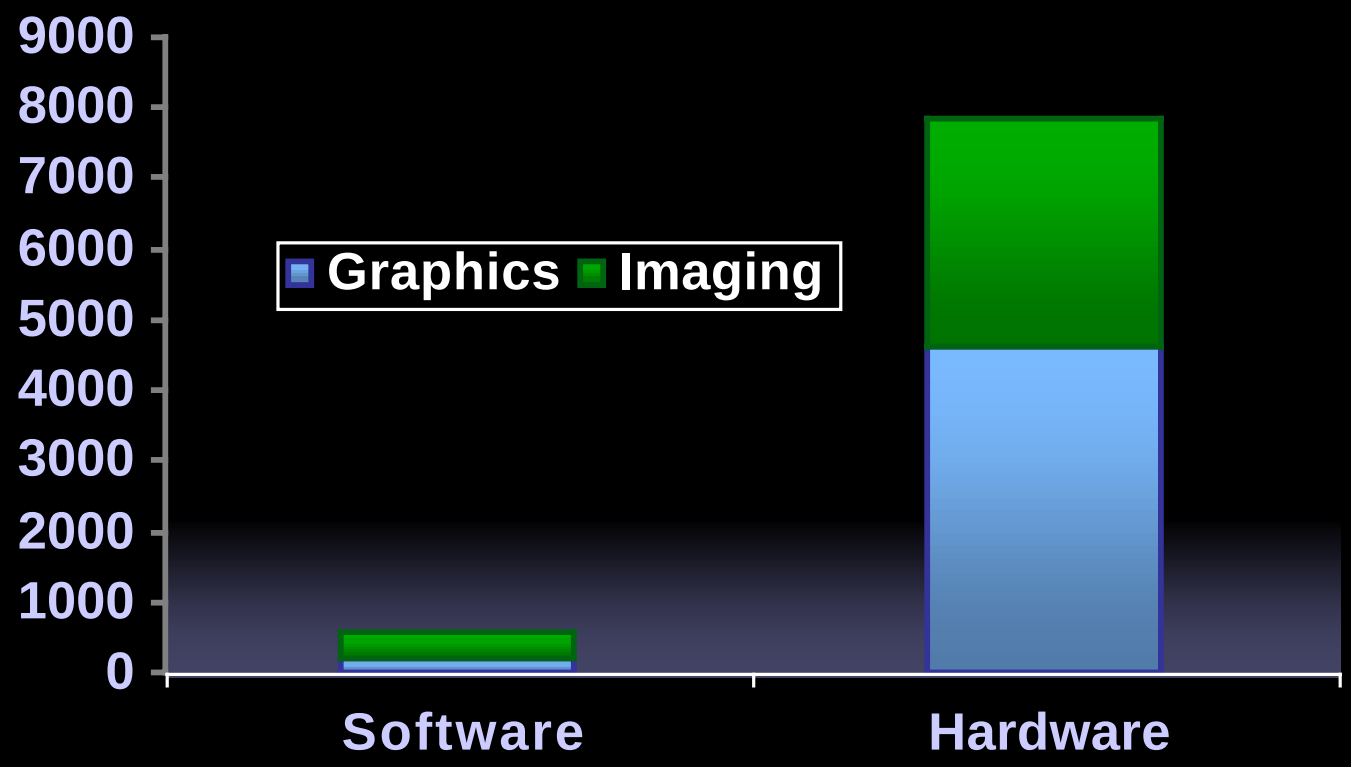
Demo



Promise Kept: Speed

Gerard Ziemski

Graphics CaffeineMarks on Mac OS X



Hardware Acceleration!!!

- Even Swing is hardware accelerated!
  - Only on Mac OS X
  - Not on Windows, Solaris

How?

- Worked with the Quartz team
- Modular architecture
- Use best available tools for job
- OS Double Buffering

What We Have Done

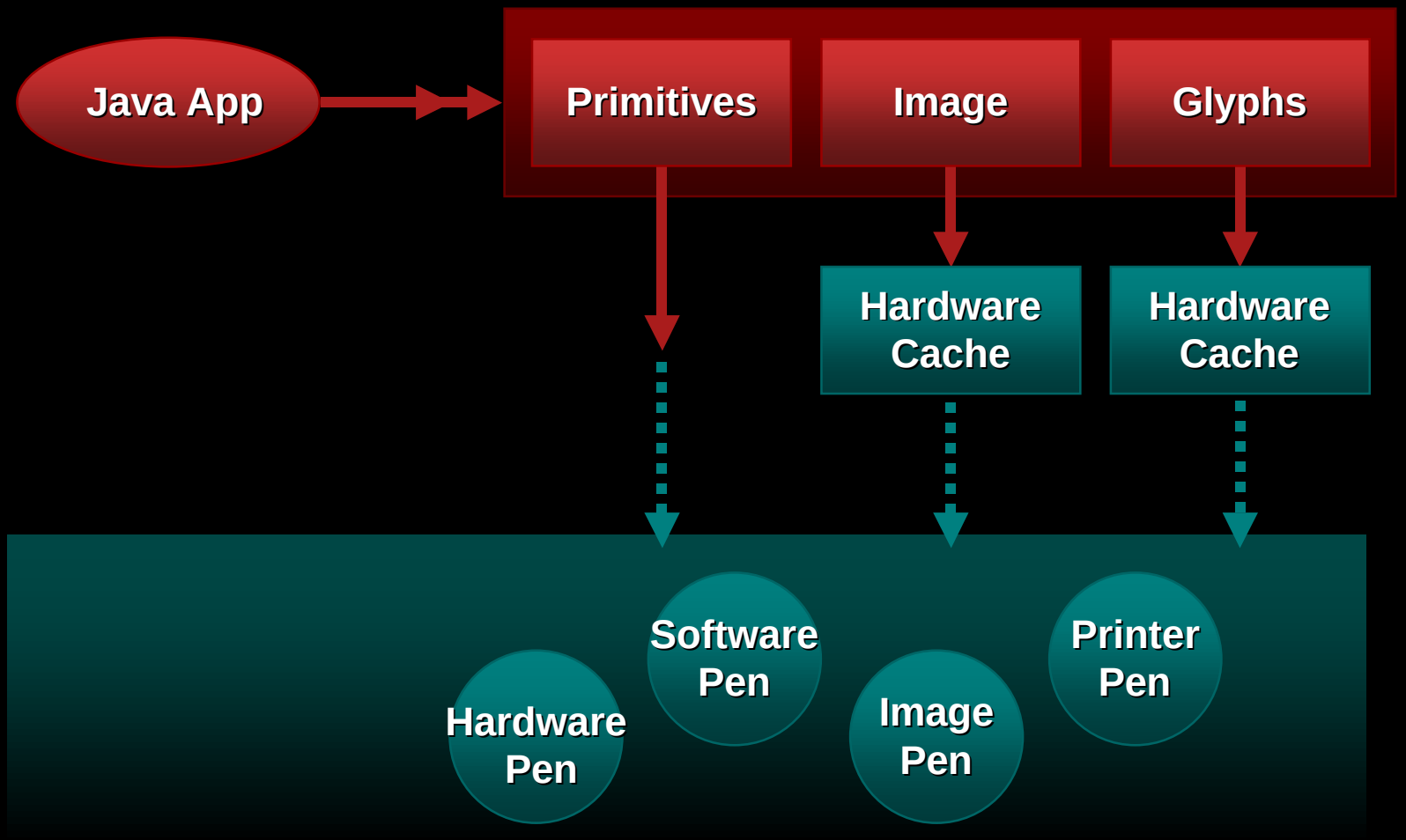
- All primitives are hardware accelerated
- 95% of all text rendering is accelerated
- General Path Tesselation being done now
- Static image upload to VRAM
- Hardware Double Buffering

Why?

- Careful Measurement
  - Profiling typical app usage
- Prototypes in late Summer 2000
  - Showed performance promise
- Project plan in Fall 2000

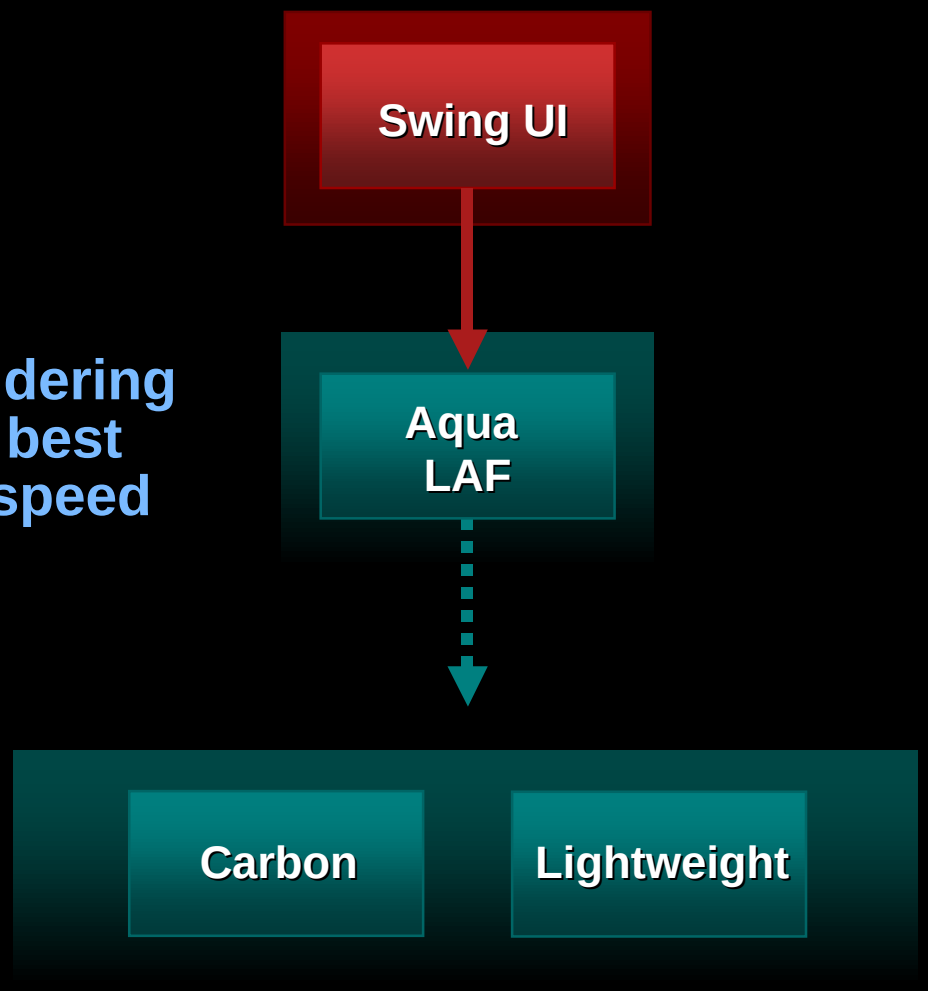
Java2D

Apple's Implementation



GUI Rendering

We use two different rendering systems for best quality and speed



Porting to Swing Has Many Advantages

- Speed
- Will always work best first
- Much richer UI framework than AWT
- Uses same rendering layer for widgets

Code

AWT to Swing

John Burkey

The Promise of Java on Mac OS X

- Java2D: Object Framework access to Quartz
- Swing: A very rich cross-platform UI framework
- Your apps run everywhere!

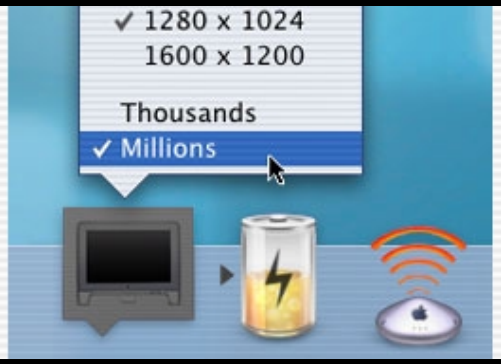
Roadmap

509 Java Security: Applet Security, J2SE Security, JSSE	Room J2 **Fri., 9:00am**
FF117 Feedback Forum: Java	Room J1 **Fri., 10:30am**

Who to Contact

Alan Samuel
Java Technology Manager
Apple Worldwide Developer Relations
bluker1@apple.com

Q&A



John Burkey and Gerard Ziemski
Java Graphics Guys
Apple Computer

http://developer.apple.com/events/urls.html

WWDC
Worldwide Developers Conference 2001